Mining Frequent ADL Patterns via Acceleration Data

ABSTRACT

Physical activity recognition is an active area in sensor mining and ubiquitous computing community. One challenge is to handle large amount of data which does not belongs to pre-defined classes. In this paper, we propose an unsupervised method to discover frequent Activity of Daily Living (ADL) patterns from acceleration data streams. Once an frequently-occurred ADL pattern is discovered, our approach learns a classifier for it. And the approach can keep digging for new classes of activity when new acceleration stream comes. Our approach can be divided into two phrases: A rough segmentation (pattern discovery) of the motion stream via topic model, and model learning with semi-supervised learning, whose training set is obtained by segmentation results from the first stage. Experiments on real life datasets show promising results on both ability to discover ADL patterns and recognizing accuracy of its learned classifiers.

Author Keywords

ADL discovery; ADL recognition; Pattern discovery

INTRODUCTION

Activities of Daily Living (ADL) recognition is an active research area in senor mining and wearable computing. Unlike "simple activity" such as "walking, sitting, running" and some given gestures, an ADL level activity such as "having dinner", "vacuuming" is composed of a series of sub-activities in a non-stable order. The recognition of ADLs such as "having dinner", "working at desk", "dressing up" provides context-based applications with a higher level abstraction of the user context. With the popularization of smart phones and wearable gadgets such as smart bracelets, the efficient discovery and recognition of ADL is able to provide useful context to applications ranging from healthy care to context-aware smartphone, from education and smart home.

The generally accepted scheme to recognize activities is to use machine learning techniques to map pre-segmented acceleration data to the corresponding pre-defined activity labels. However, the online recognition and personalized activity modeling introduces challenges that do not occur in the generally accepted offline approach. One of the challenges is that the user's actual activities may not occur in the pre-defined label list. It requires the system to automatically discover these out-of-vocabulary activity patterns, especially when the out-of-vocabulary data occupies the majority of the observed data.

In this paper, we propose an unsupervised approach to discover ADL patterns from unlabeled acceleration data. We take unlabeled acceleration stream as input and learned classifiers (one for each discovered ADL pattern) as output. Our method is based on topic model and semi-supervised learning.

By discovering frequent ADL patterns unsupervisedly, our approach adapts itself to the user's out-of-vocabulary activity patterns. Also it offers the opportunity to build a personalized model by leveraging the user's own motion data, in a way where the class of learned model and the parameters of learned model is adaptive the user's own daily routine and motion style.

The main contributions of this paper is as follows. First, it propose an unsupervised scheme to discover ADLs in acceleration stream and learned classifiers for each discovered pattern. Secondly, it applies LDA model directly on acceleration data to perform an ADL-grained segmentation.

The rest of the paper is organized as follows. Section "Approach Overview" gives an overview of our design. Section "Raw Segmentation with LDA" discusses how to segment the acceleration stream with LDA model. Section "Learning a Classifier" introduces the ADL learning process based on the earlier segmentation. Experiments are performed in Section "Experiments". Finally, in Section "Conclusion and Future Work" we summarize our results and give an outlook on future work.

RELATED WORK

Earlier human activity recognition focus on gestures and simple activities such as walking, running, going upstairs [6, 7, 8]. In 2012,Rai et al [10] recognized complex activities from acceleration data in smartphone. Their approach recognizes complex activities in a two stage scheme on data collected from smartphone over 8 weeks, and reached impressive results.

A couple of researches ease the labeling efforts by leverage techniques of semi-supervised learning and multi-instance learning. Maja et al [14] explored the performance of co-training and active learning in ADL recognition, and found that the train data can be reduced to a surprisingly small amount. Stikic et al [13] applied multi-instance learning to provides user with more abstract annotation ways.

Another line of research avoids the labeling efforts by unsupervised discovery. ADR-SPLDA [2] explored the activity discovery in the smart home with extensive sensors in a hierarchical approach. Cook et al[3] also discovered activity pattern in an smart environment by discovering frequently-occurred event sequences. Huynh et al [5] and Minnen et

al [9] detected certain simple activities from on-body acceleration data in an unsupervised way. Topic model is a popular method in unsupervised activity discovery. Typical ways to use topic model in activity discovery is a hierarchical approach. First, researchers learn low-level short activity or posture in an supervised way and convert the raw sensor stream into low-level activity sequences. Then topic model is applied on these activity sequences to discovered the high-level pattern. Huynh [4] applied topic model on low-level recognized activities to discovery daily routine pattern. ADR-SPLDA [2] also applied topic model on low-level recognized sequences. Pathak et al [15] uses the similar approach in video stream mining.

APPROACH OVERVIEW

Design Concerns
The goal of our system is to mine ADL activitie patterns from hours of on-body sensor stream unsupervisedly. The system discovers frequently-occurred ADL pattern from acceleration stream and learn a classier for each discovered pattern. User needs to manually put labels on these patterns.

We'd like our system to discover human activities in such a manner:

- **Online discovering** The system should be able to discover activities continuously from continuous sensor data. It differs from the offline approach, which trains the activity classifiers from a fixed training set. Our system should keep mining when new sensor data comes. It should recognize the known activity classes with its known classifiers and discover new activity patterns from the new data.

- **Scalable to activity pattern numbers** The number of activity class won't be manually configured in advance. The system should decide itself how many activities to model.

Framework of Our Approach
We design our system as a unsupervised learning system to meet these requirements.

We design our system as an iterative discovering process to meet these requirements. By introducing topic model, we perform pattern discovering by analyzing the distribution of topics over small motion windows. An relatively similar and steady distribution over a period of time indicates a potential activity pattern. In this way, the number of patterns won't be the number of topics. That is the main difference between our method and the K-clustering method and previous topic model based method. In previous methods, the number of patterns are pre-defined, either cluster number K_c or topic model number K_t. In our method, the number of patterns are decided at runtime.

Figure 1 illustrates the overall framework of our design. Our method could be divided into two phrases. The first phrase applies an LDA algorithm to do a rough segmentation to the raw acceleration stream. We apply the LDA estimation to documents generated over a slide window of 10-30 seconds and shifted by 10-30 seconds. Then the raw segmentation is performed by analyzing the distribution of topics of the
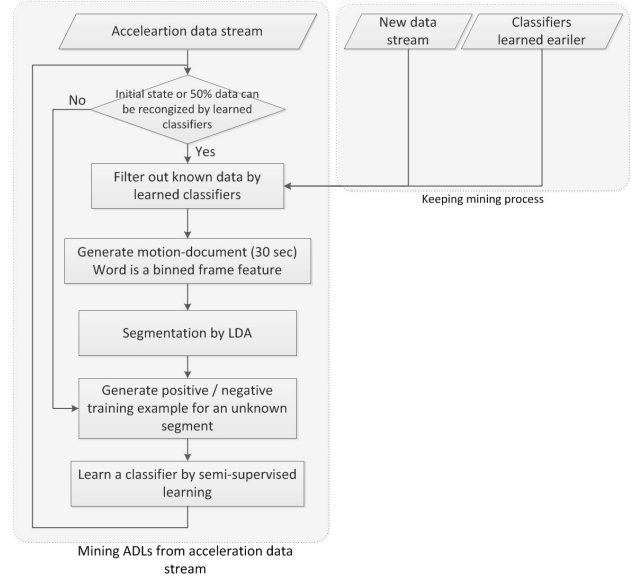

Figure 1. Framework of mining ADLs from acceleration data with LDA and semi-unsupervised learning

neighboring windows. Each segment is regarded as a potential activity pattern. The second phrase generates training examples for a certain activity from the segmentation result and learns the classifier for this activity with semi-supervised learning techniques. Phrase two has two functions. One is to try to find all activity instance who belong to the same pattern with activity instance in the target segment. The other one is that, after user put a label on the learned classifier, then this classifier could be used directly to recognize activities on new data.

The activities in the dataset are usually heavily unbalanced. Some activities (e.g. "sitting desk activities", "sleeping deeply") appears more times and lasts much longer than other activities (e.g. "fanning at barbecue"). We tackle this problem by doing the mining process iteratively. After the recognized data with learned classifiers are more than a threshold, we will filter out these recognized data and start the whole learning process all over again – from raw segmentation to classifier learning.

This scheme can adapt itself to new motion data in an intuitive way. When the new data comes, the earlier learned classifiers first filter out these known patterns and left the unknown ones. Then the remaining samples are reorganized in time-increasing order and go through the mining process like a normal stream. The recognized samples can serve as negative samples for classifier training. In this way, the scheme can keep mining new patterns from new data stream once after a couple of hours.

RAW SEGMENTATION WITH LDA
One tough problem in our unsupervised scheme is how to leverage all these unlabeled and even unsegmented data. In the experiment, we found that when we apply topic model on windows of raw data, if the neighboring windows belongs to the same ADL, then the topic distribution of these windows
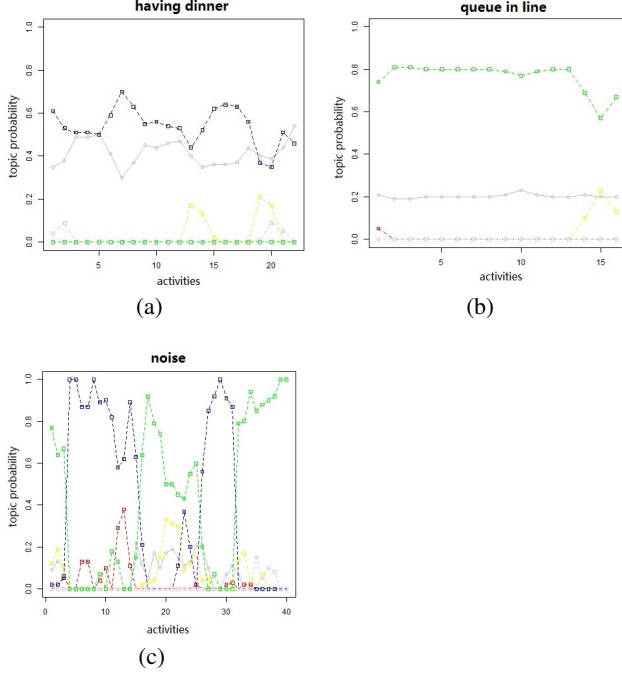
Figure 2. Topic distribution of motion windows when applying LDA to motion streams. One color represents one topic. (a) having dinner (b) queue in line (c) noise data

tends to be similar (Figure.2 (a) "having dinner", (b) "queue in line"). However, if the neighboring windows are noise data or transition data between two activities, then the topic distributions tends to be shapeless (Figure.2 (c) "noise").

Inspired by this fact, we sliced the raw acceleration data into segments by analyzing the topic distribution of neighboring windows. Windows in one segment are regarded as data of the same activity class, then can be treated as the positive examples for this class of activity.

In this section, we will first introduce briefly about LDA model, and then our ways to apply this method to raw acceleration data, and finally introduce our rough segmentation approach based on the result of LDA.

Topic Model with LDA

Topic model [1] stems from techniques in natural language processing. It regards a piece of document as a collection of words, regardless of the word's position. Each word is also related to one or some topics, thus a piece of document could be regarded as a combination of some topics. For example, a document on "The Human Genome Project" is regarded as a collection of words sampling from topic "Human Evolution", "Genetics" and "Computer Science".

Topic model can be used to generate a document automatically. More interestingly, topic model could infer the topic distribution of each document and the word distribution of each topic, given a collection of documents. The Latent Dirichlet Allocation (LDA), as a statistical model, is usually used to capture this intuition. To understand how LDA works, let's

first describe the generative process – how to generate a document given a distribution of topics.

Topic z is defined as a distribution over a fixed vocabulary. A document d is regards as a distribution over a set of topics $z_1, z_2, \ldots, z_T$. To generate a word w in a document at a certain place, we first randomly choose a topic over the topic distribution, and then random choose a word in the chosen topic over the word distribution. This process could be modeled in the formula below.

$$p(w|d) = \sum_{z=1}^{T} p(w|z)p(z|d) \tag{1}$$

LDA models the reversed process. The documents are observed, while the topic distribution over documents and word distribution over topic are yet to be inferred. LDA treats the observed documents as arising from the generative process that includes hidden variables [1], which is the hidden topic structure. LDA computes the conditional distribution of the hidden variables given the observed variables by using joint distribution. This problem could be reduced to find parameters α for the dirichlet distribution and parameters β for the topic-word distributions $p(w|z, \beta)$ that maximize the likelihood L of the data for documents [4] d = 1, ...M:

$$L(\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha)(\prod_{n=1}^{N_d} \sum_{z=1}^{T} p(w_n^d|z, \beta)p(z|\theta_d))d\theta_d \tag{2}$$

where θ_d is the topic distribution over document d and each document d consists of words $w_n^d (n = 1, 2, \ldots, N)$.

LDA assumes that the order of words does not matter. This assumption makes it suitable for exploring ADL activity structure. Because ADLs are often composed of a series of sub-motions and gestures, and they are often performed as interleaved and in an random order.

Generating Motion-Document

In this section, we will show how LDA is applied on acceleration data streams by introducing motion-documents. A motion-document is defined as a feature collection over a sliding window of 30 seconds shifted in increments of 30 seconds. Thus, raw acceleration stream over hours could be regarded as a collection of documents. Each document contains the motion information about a 30-second-long activity.

As words to a plain document, a frame-based feature is a word in our motion-document. The dataset is sub-sampled at a relatively low frequency at 2Hz, so we have 60 frames in a motion-document. To each frame, these features are extracted on each sensor channel: mean and standard deviation,which sums up into F features. Since these features are real-number features, we can not use them as a word. This problem is tackled by binning the feature value into N_{bin} equal-frequency bins. Thus, the vocabulary size is $N_{bin} * F$.

To sum up, a motion-document, representing 30-second-long motion data, has $60F$ word with 60 frames. And $N_{bin} * F$ different words compose our vocabulary.

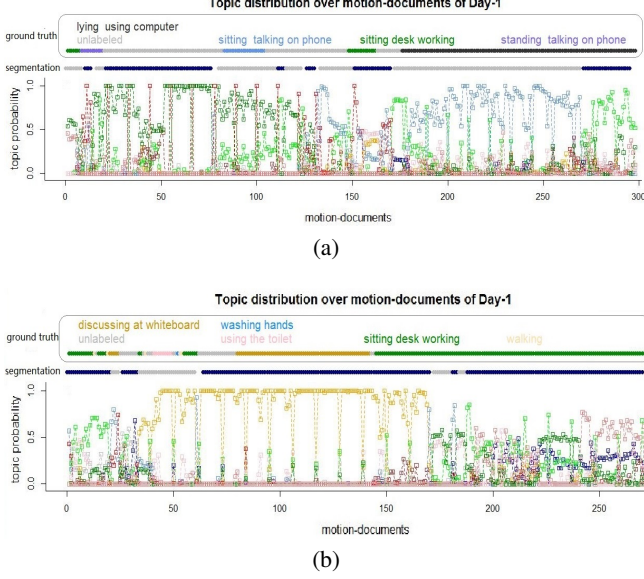### Segmentation of Acceleration Stream based on LDA

(a)

(b)

Figure 3. Topic distribution of motion-documents in Day-1

As the result of document generating procedure, we obtain a list of motion-documents representing hours of human motions. Our goal is to give a rough segmentation of these motion data, where continuous data of the same activity class is segmented into one segment. We achieve this by applying LDA model on motion-documents and segmenting based on their topic distributions.

Figure 3 shows the result of LDA model on motion-documents of two fragments in Ubicomp Day-1 data with K (K=10) topics. A point on x-axis represents a motion-document, listed in time-increasing order. y-axis represents the topic distribution over one motion-document. One line of the same color represents one topic. The topic distribution of motion document d is defined as $TD(d)$. $TD(d, k)$ refers to the kth topic's proportion over document d. For example in Figure 3 (b), motion-documents of activity "discussing at whiteboard" are almost composed by vocabulary in topic-orange, while motion documents of activity "sitting desk working" are a mixture of topic-blue, topic-green and topic-pink.

By observing Figure 3, we can find that usually topic distribution of the same activity class over a period of time tends to be similar, while the topic distribution in activity transition time tends to be noisy and shapeless. Inspired by this observation, we segment the motion-document list in the following way:

1. If neighboring n ($n = 5$) motion-documents share similar document distribution, then these n motion-documents are regarded as a mini-segment. This is done computing the similarity of topic distribution $SimTD$ over a sliding window of

Figure 4. Segmentation result of acceleration data stream in Day-1

size n and shifted by 1.

$$SimTD_{t,n} = -\sum_{k=1}^{K} sd(TD(d_t, k), TD(d_{t+1}, k), \\ TD(d_{t+2}, k), \ldots, TD(d_{t+n-1}, k)) \quad (3)$$

,where $sd()$ refers to standard deviation.

2. Merge the neighboring mini-segments together.

The proposed segmentation approach is a rough estimation of activity segmentation. Figure 4 illustrate the segmentation results of Day-1 data. Ideally, documents with the same activity class over a relatively long period of time are estimated into one segmentation (e.g. "discussing at whiteboard", "lying using computer"). However, some long activities are cut into several segments (e.g. "sitting desk working"). This may due to the interruptions by some short activities or the severe unbalance of the document set. More details will be discussed in Section "Experiments. Discussion on LDA-based Segmentation". This problem can be eased by semi-supervised learning in the next phrase.

LEARNING A CLASSIFIER

Training Samples
Now that the motion stream is already cut into segments, our goal is to learn the classifiers for ADL classes. One main decision to make is how to generate the positive and negative training samples for learning. Here, we propose a method to automatically generate training examples. First we find the longest un-recognized segmentation, and regard it as our target. Our goal is to find a classifier that can recognize all the motion samples that has the same ADL class with the samples in this target segment. Motion samples are also obtained by a slide window over 30 seconds and shifted by 30 seconds, one to one correspondent to a motion-document. Features of sample are the mean, standard deviation and entropy, energy on frequency space [10] over each channel. Ideally the classifier will not only recognized the samples in the target segment correctly, it will also identify other segments which are filled with samples with the same activity class.

Positive samples can be easily gained. We randomly sample 70% samples from the target segment as the positive training examples.

Negative samples are more complex. Balance between the aggressive and conservative need to be considered. An aggressive sampling approach may take in extensive negative samples and even true samples, which brings the classifier with high specificity but low sensitivity. On the other hand,

4

the conservative approach makes sure that the negative examples we get are true negative ones, which leads to high sensitivity but low specificity due to insufficient negative examples. To achieve the balance, we propose a method that sampling negative samples by considering the difference of topic distribution between two segments.

Difference of topic distribution between segment $S_1(d_{t_1}, d_{t_1+1}, \ldots, d_{t_1+n_1-1})$ and $S_2(d_{t_2}, d_{t_2+1}, \ldots, d_{t_2+n_2-1})$ measures the difference between two segments. It is measured by the mean proportion of each topic in the segment, which is defined as follows:

$$Diff(S_1, S_2) = \sum_{k=1}^{K} \| \frac{1}{n_1} \sum_{j=0}^{n_1-1} TD(d_{t_1+j}, k) \\ - \frac{1}{n_2} \sum_{j=0}^{n_2-1} TD(d_{t_2+j}, k) \| \quad (4)$$

The negative training samples for target segment S_{target} is obtained in the following way:

- **short segment** S: treated as noise

- **long similar segment** S $(Diff(S, S_{target}) < threshold)$: The sample S is hard to tell whether belongs to the target activity. We treat it as uncertain samples.

- **long different segment** S $(Diff(S, S_{target}) > threshold)$: The sample S behaves quite different to the target segment and could be used as negative samples. To take full advantage of its difference, we sample the negative ones proportional to $Diff(S, S_{target})$.

Learn a Classifier with Semi-Supervised Learning

Since our training examples are obtained from segmentation estimation rather than manually picked, our training examples are very limited in both number and variety. One of the best choices in learning a classifier with these limited data is to learn it with semi-supervised learning. Research on semi-supervised learning on activity recognition [14] indicates that a classifier trained with only 6.3% data by co-training performs surprisingly competitive to the one with all data. Here, we learn the classifier for the target segment with co-training technique.

Co-training is a wrapper-algorithm that repeatedly improve self by augmenting the training process with an additional source of information [14]. It is ideally trained with two views of the same data. If we had the GPS data or audio data of the same user in the same day, that would do us the best. Unfortunately, we don't. An inferior choice for us is to use two different classification algorithm. Here we use decision tree and SVM. Decision tree and SVM train two classifiers separately on the training data that we gained from segmentation, and teach another one by augmenting each other's training set with its own most confident predications on the remaining samples. In our experiment, we repeated the training and teaching iteration by four times and use the SVM classifier on the fourth iteration as the final classifier.
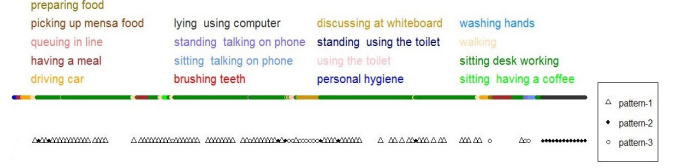

Figure 5. Three activities discovered in Day-1

Figure 5 shows the labeling results of three learned classifiers on data Ubicomp Dataset Day-1. Comparing the classes labeled by the classifiers with the ground truth, we find that our labeling result has high correlation to the ground truth, which indicates that our approach can successfully discover patterns in an ADL's granulate. Due to the high correlation, we regard these three classifiers as the model of activity "sitting desk working", "lying using computer" and "discussing at whiteboard". These three activities covers 81% samples in the day-1 dataset. And the learned classifiers achieve accuracy of 71%, 80% and 95%.

EXPERIMENTS

We use two open dataset Ubicomp08 and PAMAP in our experiment. In section "Datasets", we will give a brief introduction about these two datasets and give our parameters in the experiment. In section "Discussion on LDA-based Segmentation", a detailed discussion is given on the segmentation results. We also designed three experiments to measure our approach's ability on discovering ADL patterns (Section "ADL Discovery" and "New Patterns Discovery on New Data") and accuracy of its learned classifiers (Section "Recognize ADLs").

Datasets

Ubicomp Dataset
Ubicomp08 dataset [4] recorded the daily life of one person over a period of sixteen days, of which seven days are annotated with ADL level. Two wearable accelerometers were placed at the dominant wrist and in the right pocket (i.e. slightly below the right hip) of a single male subject as he was performing everyday activities . The accelerometers collects data at 2Hz frequency. Each day of the records consists of roughly 12 hours of data.

PAMAP Dataset
PAMAP dataset [12, 11] contains data 12 different activities (including lying, walking, vacuum cleaning, ironing, etc), performed by 9 subjects wearing three inertial sensors and a heart rate monitor. The three inertial units are worn at chest, hand and ankle, recording 3D acceleration data at 100Hz. The data is collected in realistic scenes. Altogether, over 10 hours of data is collected, of which nearly 8 hours are annotated.

Discussion on LDA-based Segmentation

By comparing our segmentation result with the ground truth annotation, we divide our segmentation result into three circumstances. Namely,

- **One segment to one activity** This is the ideal case. One ADL over a period of time is segmented to one segment. This often happens when the ADL lasts an appropriate

5

time, and its motion differs from its neighbor ADLs. This case takes up to about 30% in Ubicomp08 dataset and 60% in PAMAP dataset.

- **Multiple segment to one activity** This case often happens to heavily unbalanced data. In Ubicomp08 dataset, "sitting desk working" takes up over 70% of the time. Though it occurs at several periods of the day, one period is relatively long. Our method often sliced one "sitting desk working" period into several segments. This may due to three reasons. (1) The activity is interrupted by some short motions such as 'stand up and go for a drink'. (2) The activity is so long that makes the document set unbalanced and force the LDA to model it on a more fine granular. (3) The subject may change his posture or motion style over such long time.

- **One segment to multiple activity** This case happens when two similar ADL neighboring to each other. In PAMAP dataset, subject No.105 performs "ironing" right after "standing". The topic distribution over these two ADL is rather similar, and our system divided them into one segment. Another scenario is that one segment contains several short ADLs in an severely unbalanced dataset. This often happens at the first iteration on segmenting Ubicomp08 dataset. As discussed above, activity "sitting desk working" takes up to 70% of the whole motion-document set, and forces the LDA to model it more fine and model other activities more coarse. This problem is eased by our iterative approach. After discovering pattern "sitting desk working" in the first iteration, we filter this pattern out and start the segmentation process again on the remaining motion-documents. On the second iteration, the motion-documents are balanced and our approach are able to segment these short activities successfully.

ADL Discovery
Since our approach discovers ADLs patterns in an unsupervised way, one of the common scenes to use our approach is make it to mine hours of sensor data without any prior knowledge. Then the user could put labels on the discovered patterns. In the experiment, we applied our our mining algorithm on each day data of Ubicomp dataset and 5 subjects in PAMAP dataset. And we evaluated the performance of our method by measuring its discovering coverage and accuracy. Namely, we focus on

- **Activity count** How many ADL patterns are discovered?

- **Discovery coverage** How many samples in the dataset can be covered by the discovered patterns?

- **Recognition accuracy** How many samples in the dataset are correctively clustered?

After the system ran on the acceleration data, we compared the discovered patterns with the ground truth annotation, and attached labels for these patterns. For each discovered pattern, we only attached one most related label (sometimes "unknown" for unannotated data). The coverage, sensitivity and accuracy metrics are measured based on this label.

10 distinct activity patterns (Table 1) are discovered in seven days in Ubicomp08 dataset, including "sitting-desk-working", "discussing-at-whiteboard", "driving-bike", "fanning-in-barbecue", etc.

8 distinct activity patterns (Table 2) and three unknown patterns are discovered in PAMAP dataset, including "watching TV", "vacuum cleaning", etc.

Table 3 shows the detailed mining results on Ubicomp08 dataset. Overall the discovered patterns covers 86% of the raw data, and the overall recognizing accuracy reaches 75% sensitivity and 79% accuracy. The result shows the good adaptivity of our scheme to new activity patterns. Day-4 seems a outing barbecue day in the dataset. Some barbecue related activities appear only on that day. The approach adapts to this phenomenon by successfully discovering pattern "kneeling making fire" and "fanning at barbecue".

The accuracy is relatively better for activities that performed in a continuous period of time, e.g. "watching-movie" (Day-5), "discussing-at-whiteboard"(Day-1, Day-4). Activity "sitting-desk-working" obtains a relatively lower recognizing sensitivity and accuracy. One potential reason may be that the person performs this activity during most of the day, but his working style may differ from morning to afternoon. Some classifiers detects his morning desk working sensitively but ignores his afternoon desk working. This phenomenon occurs on Day-6. Three pattern are discovered, namely corresponding to "sitting-desk-working" at in the morning, in the afternoon and in the evening. Another reason is that although desk working occupies most of the time, it is interrupted by many other activities. Therefore the segment for desk working is not very long and we don't have sufficient positive examples for training the classifier.

Discovered ADL Patterns	
Sitting desk working	Lying using computer
Discussing at whiteboard	Driving bike
Driving car	Having a meal
kneeling making fire	Attending a meeting
Fanning at barbecue	Watching movie

Table 1. ADL patterns discovered in Ubicomp08 dataset

Discovered ADL Patterns	
Nordic walking	Walking
Computer Work	Watching TV
Vacuum cleaning	Ironing
Sitting	Standing

Table 2. ADL patterns discovered in PAMAP dataset

New Patterns Discovery on New Data
Our system is an online learning system, it can keep taking in new data and exploring new activity classes once after a while. When new sensor data comes, the learned classifiers will first recognize samples in the new dataset and attach labels for them, and then we will keep on mining the remaining data for new activity classes. The remaining data makes

up the new dataset, and go through stages of segmentation and learning. Here, we test our algorithm of the ability of discovering new activity classes with new data, with some old learned classifiers. We experimented on the Ubicomp08 dataset. Classifiers learned in day-1 are treated as known models. Day-2 and Day-4 data are treated as new data. After recognizing three known activities, we also discovered new patterns corresponding to "having a meal" "driving bike" in day-2 and "kneeling making fire" "fanning barbecue" in day-4.

Recognize ADLs

Since a classifier is learnt for each discovered pattern, this classifier could be used to classify new data.

Here we did an experiment to explore the classifying performance of our learnt models. We also experimented this on Ubicomp08 dataset. For seven days, we discovered patterns on each day data and manually attach the labels to the classifier by comparing the classification results to the ground truth annotation. As usually, only one label is attached to one classifier. Then we tested these classifiers on the other six days of data.

Table 4 shows the recognizing accuracy of some frequently occurred activities. The overall recognizing performance reaches 69% sensitivity and 78% accuracy.

		Coverage	Sensitivity	Accuracy
D1	Sitting desk working	0.69	0.68	0.71
	Lying using computer	0.08	1.00	0.80
	Discussing at whiteboard	0.04	1.00	0.95
	Average	**0.81**	**0.89**	**0.82**
D2	Sitting desk working	0.75	0.67	0.70
	Attending a meeting	0.08	0.25	0.92
	Driving bike	0.04	1.00	0.94
	Average	**0.87**	**0.64**	**0.85**
D3	Sitting desk working	0.79	0.63	0.67
	Having a meal	0.05	0.49	0.72
	Average	**0.84**	**0.56**	**0.70**
D4	Sitting desk working	0.66	0.63	0.74
	Discussing at whiteboard	0.07	0.99	0.73
	Kneeling making fire	0.02	0.95	0.95
	Walking	0.05	0.38	0.92
	Fanning at barbecue	0.03	0.98	0.83
	Average	**0.83**	**0.79**	**0.83**
D5	Sitting desk working-1,2	0.84	0.84	0.57
	Watching movie	0.08	1.00	0.75
	Having a meal	0.05	0.72	0.85
	Average	**0.87**	**0.85**	**0.72**
D6	Sitting desk working-1,2,3	0.83	0.97	0.93
	Driving car	0.06	0.67	0.86
	Average	**0.89**	**0.82**	**0.90**
D7	Lying using computer	0.23	0.95	0.77
	Sitting desk working	0.64	0.39	0.60
	Average	**0.87**	**0.67**	**0.69**
	Summary	**0.86**	**0.75**	**0.79**

Table 3. Discovered ADL patterns in seven days

	Sensitivity	Accuracy
Sitting desk working	0.64	0.67
Discussing at whiteboard	0.97	0.76
Having a meal	0.46	0.70
Driving car	0.92	0.86
Walking	0.45	0.93
Average	**0.69**	**0.78**

Table 4. Recognizing performance of learned classifiers

CONCLUSION AND FUTURE WORK

In this paper, an unsupervised approach is proposed to discover and recognize ADLs from acceleration streams from wearable devices. It tackles the weakness of no labeled training data by obtaining positive/negative training data from LDA-based segmentation. The LDA segmentation not only cuts the raw stream into segments roughly, it also measures the similarity between segments based on the distribution of topics. Experiments on Ubicomp dataset indicates the promising discovering ability and recognizing accuracy of our scheme. The experiment was done on a daily bases. Ten ADL patterns are discovered and coverers 86% of the data. The learned classifiers reaches 75% sensitivity and 79% accuracy on the same day and 69% sensitivity and 78% accuracy on other days.

However, our mining approach is unable to discover patterns for short activities such as "washing hands" due to its segmentation policy. These short activity usually lasts for 2-3 motion-documents long and thus will be treated as noise. Another shortcoming is that we only mines the data on a daily bases once a time. A larger dataset (e.g. multiple days) are not helping due to two reasons. First, the computer has insufficient memory to model a larger dataset with LDA. Second,

since we obtain the positive training data from one long segment, multiple days of data won't make the segments longer and thus won't enlarge variety of positive training data.

In the future, we plan to enhance mining scheme in the following manner. First, we plan to enhance the learned classifier iteratively in its online mining scheme. When new unlabeled data comes, the learned classifier may able to leverage the new data and enhance its recognizing sensitivity. Second, we plan to offer the user an intervention scheme by offering a coarse-grain annotation interface.

REFERENCES

1. Blei, D. M. Probabilistic topic models. *Communications of the ACM 55*, 4 (2012), 77–84.

2. Chikhaoui, B., Wang, S., and Pigot, H. Adr-splda: Activity discovery and recognition by combining sequential patterns and latent dirichlet allocation. *Pervasive and Mobile Computing 8*, 6 (2012), 845–862.

3. Cook, D. J., Krishnan, N. C., and Rashidi, P. Activity discovery and activity recognition: A new partnership. *Cybernetics, IEEE Transactions on 43*, 3 (2013), 820–828.

4. Huynh, T., Fritz, M., and Schiele, B. Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on Ubiquitous computing*, ACM (2008), 10–19.

5. Huynh, T., and Schiele, B. Unsupervised discovery of structure in activity data using multiple eigenspaces. In *Location-and Context-Awareness*. Springer, 2006, 151–167.

6. Kwapisz, J. R., Weiss, G. M., and Moore, S. A. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter 12*, 2 (2011), 74–82.

7. Liang, R.-H., and Ouhyoung, M. A real-time continuous gesture recognition system for sign language. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, IEEE (1998), 558–567.

8. Mantyjarvi, J., Himberg, J., and Seppanen, T. Recognizing human motion with multiple acceleration sensors. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, vol. 2, IEEE (2001), 747–752.

9. Minnen, D., Starner, T., Essa, I., and Isbell, C. Discovering characteristic actions from on-body sensor data. In *Wearable computers, 2006 10th IEEE international symposium on*, IEEE (2006), 11–18.

10. Rai, A., Yan, Z., Chakraborty, D., Wijaya, T. K., and Aberer, K. Mining complex activities in the wild via a single smartphone accelerometer. In *Proceedings of the Sixth International Workshop on Knowledge Discovery from Sensor Data*, ACM (2012), 43–51.

11. Reiss, A., and Stricker, D. Creating and benchmarking a new dataset for physical activity monitoring. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, ACM (2012), 40.

12. Reiss, A., and Stricker, D. Introducing a new benchmarked dataset for activity monitoring. In *Wearable Computers (ISWC), 2012 16th International Symposium on*, IEEE (2012), 108–109.

13. Stikic, M., and Schiele, B. Activity recognition from sparsely labeled data using multi-instance learning. In *Location and Context Awareness*. Springer, 2009, 156–173.

14. Stikic, M., Van Laerhoven, K., and Schiele, B. Exploring semi-supervised and active learning for activity recognition. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on*, IEEE (2008), 81–88.

15. Wang, Y., and Mori, G. Human action recognition by semilatent topic models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 31*, 10 (2009), 1762–1774.